Proposed COMAL Standard

The provision of a Library facility

Throughout this document the word 'procedure' applies to both procedures and functions.

Introduction

It has been suggested that an extension to COMAL should enable a user to access COMAL procedures contained in separate files. A syntax is suggested below as well as details of implementation and possible problems. The ground covered by this feature could perhaps be combined with the use of external procedures. However one of the prime objectives has been to keep the libraries as simple to use as possible.

The LIBRARY syntax/semantics

It is proposed that a 'LIBRARY' statement or command is introduced into the COMAL extensions to the kernel. This would take the form of a declaration statement of the following syntax:

LIBRARY <file name> {,<file name>} <eol>

where file name, as defined in the COMAL 80 standard, refers to a COMAL program file.

The LIBRARY statement may occur at any place within a COMAL program except within procedures or functions. It declares a list of COMAL program files which may be searched to find a procedure in the following manner:

When a procedure call is encountered, the interpreter will search through the static procedures present in the COMAL program. If the required procedure is not found, and the LIBRARY command is currently 'active', the interpreter will then search for the procedure in the files listed after the LIBRARY keyword, in the order listed. If the procedure is located within a file, then it is loaded into an available area of the workspace and executed as a normal procedure.

This library facility of searching files for a procedure is 'activated' when a LIBRARY statement or command is encountered, and remains active until a NOLIBRARY statement or command is executed.

Successive encounters of library commands replace the list of files which will be searched.

The LIBRARY file format

The files which are specified as LIBRARY files may be any COMAL program file. All other language elements of the program file, except procedure definitions, are ignored. This means comments surrounding a procedure in a LIBRARY file will not be transferred with the procedure. Also it allows an existing complete COMAL program which contains a useful set of procedures to be used as a LIBRARY source for other programs. This is in the spirit of making

it easy for the user to create, and use, library files.

Error handling in Library Procedures

The question arises of what should happen if an error occurs when a procedure from the Library file is executing. Firstly, the procedure being executed should be an ordinary COMAL procedure and thus syntactically correct. It will be necessary to check the procedure for structural correctness before it is executed as a Library procedure. It is proposed that any errors generated while attempting to execute the external procedure would cause the interpreter to produce an error message of the following type:

```
Division by zero at line 1020
in Library file 'external'

at line 2000
in Main program
```

Only the most nested library file error would be given, to prevent the possibility of long unhelpful errors produced if nested library procedures are used. The interpreter would be left editing the main program; to correct the Library file it would be necessary to load it in and edit it separately.

Discussion

One of the major problems that implementors have to face with library procedures is the scope of variables within them. It is easier from an implementation point of view to force all library procedures to be CLOSED by default. Thus all variables to be used by them must either be passed in the parameter block, or IMPORTed into the procedure. However it would be much better from the user's point of view if he/she could define the procedures to be either open or closed by their procedure headings in the library file. This point will need discussion at the conference although I would prefer the user having the choice.